Tendency of Restaurant Reviews Mining Model Combining FP-Tree Algorithm

Qixiang WANG, Chen LI, Guanyang HE, Xinyi XU, Jing LI

School of Software Engineering, Beijing University of Post and Telecommunications, Beijing, 102209, China

email: highway2hell@163.com

Keywords: FP-Tree; Restaurant comment; Emotion tendency; UGC; POS template

Abstract. A restaurant review tendentious mining method that combines FP-Tree algorithm is presented to dig and identify high-frequency noun pairs in reviews associated with services, the environment, food, to name just a few, by mining association rules. A feature word dictionary is established and Training the training set for sentence template which is form of Part of Speech tags and match the comments with different sentence template to calculate with feature score and total score. This idea can also be used to analyze product features and areas in which UGC score calculation is needed.

Introduction

With the rising of life information exchange platforms including Dianping.com and Koubei.com, people have been paying growing attention to reviews and scores of merchants and goods for their own consumption reference. With increasing interactions between the Internet and life, reviews concerning merchants and goods present an astonishing growth trend. Confronted with the increasingly large number of reviews, it has been more and more difficult for users to extract useful information for making tendentious judgments. Therefore, computers doing a semantic analysis of large review texts and drawing conclusions about the overall emotional tendencies and emotional tendencies of each entry attribute provides good reference value for consumers to select goods or for merchants to improve their products.

Literature [1] proposes a method for restaurant reviews mining based on semantic polarity analysis, but it depends too much on tagging parts of speech such as nouns and adjectives when extracting feature words; its computational formula is too simple, and it fails to take into in-depth consideration the impact of modifiers on emotional tendencies. Literature [2] presents a tendency judging system for commendatory and derogatory texts based on keyword templates, does more planning for the emotional tendency computation of texts and explores the influential factors of modifiers on emotion. However, the matching templates are mostly designed artificially rather than generated automatically. Further, due to too few reviews of the training set, there are unsatisfied testing results of the training set and it did not make an in-depth investigation of how to optimize the matching template. Instead, it merely considers the case that the current sentence template is part of the existing template ($A \subseteq B$), without taking into account the relationship of $A \subsetneq B$ existing among most templates, or different results incurred by the sequence of parts of speech in the templates. In this case, it will cause a lot of invalid matches, which have not only reduced the utilization of the reviews, but also led to inaccurate scores.

In this study, the following emotion analysis mining tasks are devised focusing on the product features in Chinese reviews.

(1) Extract the product attribute concerned by the user;

(2) Identify the review statements in the product attribute, judge whether the emotional tendency in sentences is positive or negative;

(3) Mark the product in regard to its overall and pervasive properties

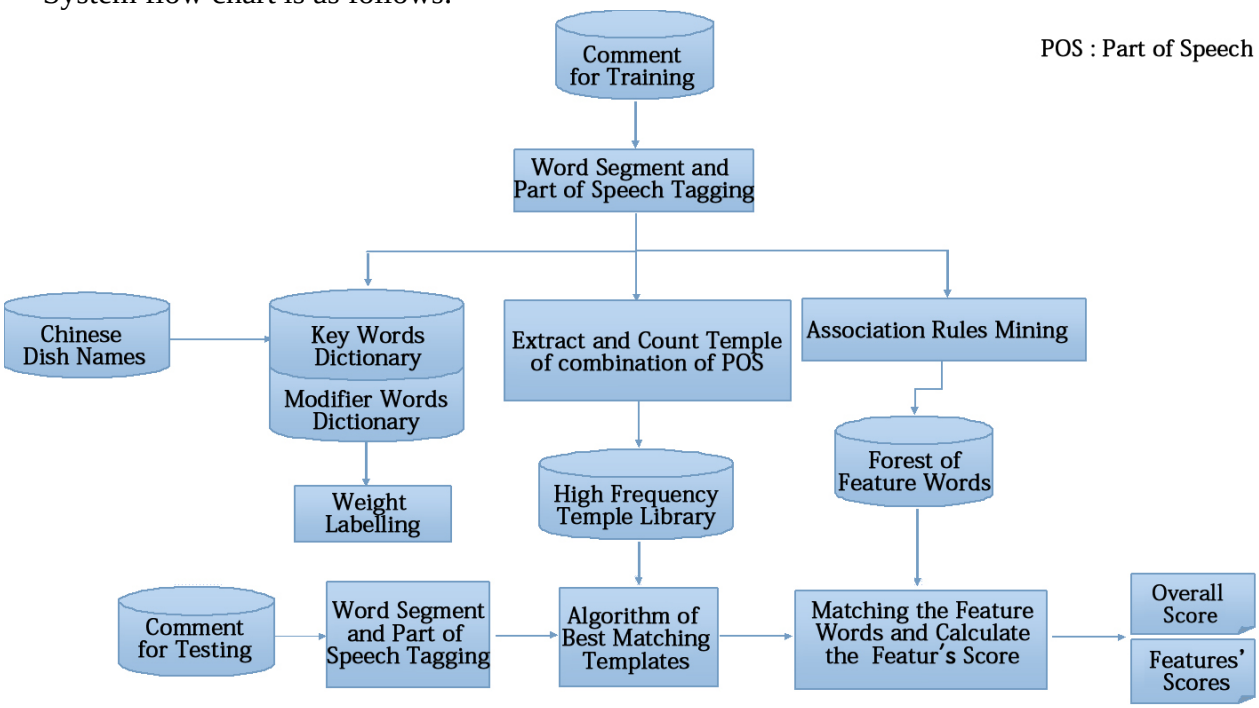
This paper specifically explores the emotional tendencies of restaurant reviews by using association rules and automatically building sub-feature feature words. When analyzing the

tendencies, it is feasible to make more detailed ratings focusing on distinctive features. Besides, building a brand new text appraise system and making a thorough study of the optimal matching template algorithm can maximize the utilization of the reviews and make the analysis more accurate.

Algorithm Corpus and the Overall Process

The corpus in this experiment is from http://dianping.com. A total of 2200 reviews from 100 on-site stores are downloaded, half for training and half for testing.

System flow chart is as follows:



Here are the algorithm steps:

1) First, mark the word segmentation and parts of speech of the training set reviews; build a keyword dictionary and a modifier dictionary.

2) Use the association rule mining algorithm to build several types of feature words interrelated in the reviews and to build a feature word thesaurus, and to make a statistic of high-frequency speech templates.

3) Label the weights of words in the keyword dictionary.

Preprocess reviews of scoring restaurants; process new words in it and match the optimal template; calculate every score and total score of different features of the restaurant.

Chinese Word Segmentation and Part of Speech Tagging

Ansi is an effective and accurate word segment tool, which can quickly split many sentences into words at once.

First, we use Ansi to split sentences into words and provide those words with POS tags. Then, add all words and their POS tags into the keyword dictionary. After that, we add all Chinese Dishes in [3] ref into the keyword dictionary as noun.

For example:

Original:服务挺好，服务员很积极，就是味道一般(The service is not half bad, and the attitude of the waiter is very gratifying, but the taste is ordinary)

After POS tagging:[服务(service) /vn 挺(quite) /d 好(good) /a , /w 服务员(waiter) /n 很(very) /d 积极(positive) /ad , /w 就(just) /d 是(is) /v 味道(taste) /n 一般(ordinary) /a]

Extraction and Statistical Process of Sentence Template

After POS tagging of 2200 training remarks, we split those remarks into short sentences by separate punctuation, and then extract sentence template and calculate the frequency of each template.

For example:

Original:服务挺好，服务员很积极，就是味道一般(The service is not half bad, and the attitude of the waiter is very gratifying, but the taste is ordinary)

We divide it into three short sentence:① 服务(service) /vn 挺(quite) /d 好(good) /a ② 服务员(waiter) /n 很(very) /d 积极(positive) /ad ③ 就(just) /d 是(is) /v 味道(taste) /n 一般(ordinary) /a

Then we can get three templates, each showed once:

① vn, d, a; ② n, d, ad; ③ d, v, n, a.

After extraction and calculation, we select the 100 most frequent templates and put them into the high-frequency template library. And the template with higher repeat frequency has more prior.

During the Process of Scoring restaurant, we just consider those templates in the high-frequency template library in order to avoid calculating useless sentences.

Associate Rule Mining

We input those short sentences into FPtree, get the maximum-relational-grade feature phrase library on service, environment, price and food, and then screening those feature phrases.

Since the network comments are written by amateurs and info mining on them is a field with large noise which involved a number of features, and the features generated by iteration are quite random, we choose FPtree algorithm based on associate rules as the method of feature mining:

a) Calculate the emotion tendency of each short sentences, and store the results in the user idea data base.

b) Calculate the word frequency of nouns after POS tagging, and use thesauri to classify those synonyms, then we get set I1.

c) Select the frequent words in I1, use them as sub feature set I1'.

d) Use the comments after POS tagging to create the associate rule transaction file.

e) Use FPtree algorithm based on associate rule to find the frequent item set between the first degree feature set I0' and sub feature set I1', regard the frequent item set as candidate feature set I.

f) Use neighboring rules to select the frequent item set I, and get the final feature set I which has two-stage features.

Neighboring Rules is:

In the Chinese comments, assuming that f is a frequent items set, and f includes n nouns or noun phrases, and s is a sentence which contains f, and the order of those words is w1,w2,...,wn. If any distance between wi and wi+1 is not exceed 2 words, we can regard f is neighboring in s.

We can get the relation of nouns using this method, which usually represents feature hierarchy. For example: 牛肉(Beef)-口感(Taste), 烹调(Cook)-口感(Taste).

Then we combine the word library with the results of FPtree, and screen them on a small scale. Finally, we get a feature library. For example: (环境)Environment includes 内部环境(The internal environment) and 外部环境(The external environment), 内部环境(The internal environment) includes 用餐区域(Dining Area), 厨房(kitchen),etc.

We represent those relationships in this form:

Belong[“店内环境”(The internal environment)]= “环境”(Environment)

Belong[“用餐区域”(Dining area)]= “店内环境”(The internal environment)

Belong[“餐厅”(Dining room)]= “用餐区域”(Dining area)

Belong[“包厢”(Balcony)]= “用餐区域”(Dining area)

In all, we get a feature library with 105 features and 3500 feature words.

Weight Marking

We assign 1 to the weight of dishes,-3~+3 to adjectives and adverbs, -1~+1 to conjunction.

Comment Pretreatment

We split the comments of the aim restaurant into words and give them POS tag, add those words which haven't showed into the keyword dictionary, and then we extract the templates of those comments.

Optimal Match of Sentence Template

1) Outline

We do further study on the algorithm on Optimal Match of Sentence Template and clarify fuzzy previous description about how to match the best sentence.

We think match has two types:

① Complete matching

It means that the model of POS tags in the short sentence is just one of the templates, and the number of words in the sentence is same as the template. For example:

Original：环境比较老(The environment is rather antique)

After POS tagging: 环境(environment) /n 比较(rather) /d 老(old) /a

Complete matching template: n,d,a

② Incomplete matching

It means that the model of POS tags in the short sentence has not a complete matching template, but has some relevant template.

In this case, we choose the higher prior template which has the same order of words and the maximum of matching words.

For example:

Original: 骨头的量貌似没以前足了(The amount of rib seems not equal to before)

After POS tagging:骨头(rib) /n 的(of) /uj 量(amount)/n 貌似(seems)/v 没(not) /d 以前(before)/f 足(equal) /a 了(meaningless)/ul

After optimal match, it matches 14th template: n,uj,n,d,a.

骨头(rib) /n 的(of) /nj 量(amount) /n 没(not) /d 足(equal) /a

It means that the word combination could not match any existing template, while having an intersection with several of them. Following is an example.

The original sentence is 骨头的量貌似没以前足了.After word segmentation and parting of speech tagging, it turns to 骨头/n 的/uj 量/n 貌似/v 没 /d 以前/f 足 /a 了/ul. And the 14th template is the most appropriate, which is n,uj,n,d,a, corresponding to 骨头/n 的/nj 量/n 没/d 足 /a.

At last, on the basis of ensuring the combination order, the degree of matching template is 5. In this case, we should select the top one in the high-frequency template library from the templates which have the same matching degree.

2) Special case: The implementation of matching algorithm in the case of incomplete matching

In order to find out the most appropriate template which could match comment A, for each template Bi, we could use the method of Dynamic programming to find out the longest common sub sequence Ci between Bi and A. Then choose the longest Ci. If there are several longest Ci, we need to choose the top one in the high-frequency template library from them.

Use arrays A[j] and B[j] to store the part of speech at index j（j=1,2,3..）, such as A[1]=’n’, A[2]=’d’…… Use La to store the length of list A, and use Lb to store the length of list B. Use a two-dimensional array to store the length of a longest common subsequence before index m of A and index n of B.

Following is the equation using dynamic programming:

```
for(k=0;k<La;k++){
    for(p=0;p<Lb;p++){
        if(A[k]!=B[p]){
            dp[k+1][p+1]=max(dp[k][p+1],dp[k+1][p]);
            }
            else dp[k+1][p+1]=dp[k][p]+1;
    }
}
```

Compare each template in the library, and update the longest common subsequence, and select the top one in the high-frequency template library. And, on the basis of the acquired array dp, you can back out the longest common subsequence.

The time complexity of the algorithm can be calculated by this expression, (the number of templates in the library) * O(The maximum length of the template * The length

Use nt to express the number of templates in the library, and use ml to express the maximum length of the template, and take time complexity under consideration, 100*O(10*10)=O(10^4). It is quite fast.

The match and scoring of feature

First, for each word in the comments which have already been matched, we need to find the position of the feature word. According to our high-frequency template library, we can see that, in most cases, a feature word often appears early in a sentence, while the modifier for it often appears behind it. Therefore, when we find a feature word, we should search behind its position for its modifier. And use the modifier to calculate the score of this feature. For example,

Secondly, use this result to adjust the score of its parent. For example,

At last, on the basis of the completion of the above, the total score of this restaurant and the score of each feature can be output. Using the algorithm in this paper, the more comments, the more features, the more detailed.

The Match and Scoring Of Feature

First, for each word in the comments which have already been matched, we need to find the position of the feature word. According to our high-frequency template library, we can see that, in most cases, a feature word often appears early in a sentence, while the modifier for it often appears behind it. Therefore, when we find a feature word, we should search behind its position for its modifier. And use the modifier to calculate the score of this feature.

For example,

骨头/n(score:1) 很/d(score:2) 好吃/a.(score:2)

$$M=1*2*2=4$$

骨头/n(score:1) 不是/d(score:-0.5) 很/d(score:2) 好吃/a.(score:2)

$$M=1*(-0.5)*2*2=-2$$

Our work has shown that, in Chinese, negative adverb only represents half the tendency of the same sentence without it. The equation is:

$$S=V_f*V_{d1}*V_{d2}*\ldots*V_{dn}*V_{a1}*V_{a2}*\ldots*V_{an}$$

While S is final score，V_f is value of the feature word，which is mostly noun. $V_{d1}\ldots V_{dn}$ is value of adverb appeared in the template and $V_{a1}\ldots V_{an}$ is value of adjective appeared in the template. The value of sub feature is the average of the score of its feature word happened to be in the user's comment.

Secondly, to calculate the value of parent feature, we take the score of its sub features 40% and the score from its own feature words 60%.

At last, on the basis of the completion of the above, the total score of this restaurant and the score of each feature can be output. Using the algorithm in this paper, the more comments, the more features, the more detailed.

Test Results

We have chosen 2200+ comments as our test set, using this system to analyse one resuaurant, we can get the following result.

```
服务：1.25
    服务员：1.75
    经理：0.75

环境：0.99
    店内环境 0.987635
        用餐区域：0.94725
        卫生区域：-0.714285
        厨房区域：0.992188
        店内气氛：2.54751
    店外环境：0.984375

菜品：1.50
    中餐：1.96875
    食材：1.936523
    调料：1.375
    饮品：0.75

价格：0.5
    服务费：0.5
```

Comparing with the result in dianping.org, we can find that it is basically In line with our expectations. Food and Service part is quite good but the environment is not quite satisfied, toilet score is even a negative number. After that we chose 11people to read 1000 comments in the 2200 ones and calculate the accuracy rate with our subjective feelings.

	Food	Environment	Service	Price
Accuracy rate	73%	81%	83%	90%

While the food area is quite a big one with the amount of almost 3/4 features and the price area is quite simple. The complexity may cause the accuracy difference in each feature field.

Special statement

This paper is funded by the Beijing University of Posts and Telecommunications innovation base.

References

[1] Turney P, Littman M. Measuring praise and criticism: Inference of semantic orientation from association[J]. ACM transactions on Information Systems, 2003, 21(4): 315-346

[2] Yu Pan，Hongfei Lin，Restaurant reviews mining based on semantic polarity analysis[J]. Journal Computer Engineering，2008,34(17):208-210.

[3] Fanbo Meng，Lianhong Cai，Bin Chen，Peng Wu，Research on text appraisable system，Journal of Chinese Computer Systems，2009,30(7):1458-1461.

[4] Lin Sun，Yaping Gao，Guojie Song. A new data mining model and algorithm [J]. Journal of Zhengzhou University of Light Industry (NATURAL SCIENCE EDITION)，2001,16(4):35-38.

[5] Jiawei Han, Micheline Kamber. Data mining Concept and Techonology[M]. Zeming Fan. Beijing：Machinery Industry Press，2001.

[6] HIT Synonym Forest Dictionery. http://ir.hit.edu.cn

[7] Beijing Tourism Bureau，Translation of Chinese dishes. http://www.langtech.org

[8] Dazhonngdiangping. http://www.dianping.org.